Refine Search

Search Results -

Terms	Documents
705/62	145

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search:

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Saturday, March 27, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L27	705/62	145	L27
L26	705/10	2190	L26
L25	705.clas.	25964	L25
L24	701.clas.	28234	L24
L23	701/208	1844	L23
L22	701/207	2324	L22
L21	709/217	4963	L21
L20	709.clas.	27424	L20
L19	707/10	8002	L19
L18	707.clas.	19610	L18
L17	707/104.1	3769	L17
L16	707/7	1448	L16
L15	707/6	2403	L15
L14	707/5	2847	L14

L13	707/1	6161	L13
L12	707/4	3482	L12
L11	707/3	6029	L11
L10	L8 and geocoded near(database or data with base)	15	L10
L9	L8 and (database or data with base)	259	L9
L8	L6 and (anchor or point or star)	273	L8
L7	L6 and anchor	13	L7
L6	L3 and (radials or lines or centroids or points)	287	L6
L5	L4 and anchor	0	L5
L4	L3 and radials	18	L4
L3	geocod$	332	L3
L2	L1 and geocod$	4	L2
L1	anchor near4 point	14708	L1

END OF SEARCH HISTORY

First Hit Fwd Refs

Generate Collection Print

L10: Entry 10 of 15 File: USPT Jul 24, 2001

US-PAT-NO: 6266612
DOCUMENT-IDENTIFIER: US 6266612 B1

TITLE: Position based personal digital assistant

DATE-ISSUED: July 24, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Dussell; William O.	Pescardero	CA		
Janky; James M.	Los Altos	CA		
Schipper; John F.	Palo Alto	CA		
Cowl; David J.	Sunnyvale	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Trimble Navigation Limited	Sunnyvale	CA			02

APPL-NO: 09/ 334521 [PALM]
DATE FILED: June 16, 1999

PARENT-CASE:
RELATED APPLICATION This application is a Continuation of application Ser. No. 08/738,938 filed Oct. 24, 1996, now U.S. Pat. No. 5,938,721 issued Aug. 17, 1999.

INT-CL: [07] G01 S 5/02

US-CL-ISSUED: 701/207; 701/211, 701/213, 342/357.17
US-CL-CURRENT: 701/207; 342/357.17, 701/211, 701/213

FIELD-OF-SEARCH: 701/211, 701/213, 701/1, 701/207, 342/357.06, 342/357.1, 342/357.17

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
5032083	July 1991	Friedman	434/112
5311194	May 1994	Brown	342/357
5444444	August 1995	Ross	340/994

☐	5457629	October 1995	Miller et al.	364/424.01
☐	5470233	November 1995	Fruchterman et al.	434/112
☐	5528248	June 1996	Steiner et al.	342/357
☐	5559707	September 1996	DeLorme et al.	364/443
☐	5576687	November 1996	Blank et al.	340/438
☐	5646629	July 1997	Loomis et al.	342/357
☐	5682525	October 1997	Bouve et al.	395/615
☐	5699244	December 1997	Clark, Jr. et al.	364/420
☐	5732074	March 1998	Spaur et al.	370/313
☐	5790974	August 1998	Tognazzini	701/204

ART-UNIT: 361

PRIMARY-EXAMINER: Zanelli; Michael J.

ATTY-AGENT-FIRM: Blakely, Sokoloff, Taylor & Zafman LLP

ABSTRACT:

A task description is stored in a database accessible by a mobile computer system. The mobile computer system receives positioning information corresponding to its geographic location and indexes the database based on the positioning information when the information indicates that the mobile computer system is in a geographic location that facilitates completion of a task associated with the task description. The database may be resident in the mobile computer system or accessible in other ways, for example, via the Internet. The task description preferably includes a geocode which corresponds to the geographic location at which completion of the task may be facilitated. The task description may also include textual, voice or other message which can be displayed and/or played back to a user. The positioning information may be obtained from a GPS satellite, a GLONASS satellite or a pseudolite. The mobile computer system may be a portable unit, such as a PDA, or integrated within a vehicle.

36 Claims, 2 Drawing figures

First Hit Fwd Refs

Generate Collection Print

L10: Entry 12 of 15 File: USPT Nov 30, 1999

US-PAT-NO: 5995947
DOCUMENT-IDENTIFIER: US 5995947 A

TITLE: Interactive mortgage and loan information and real-time trading system

DATE-ISSUED: November 30, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Fraser; Stephen K.	Livermore	CA		
Adiga; Sadashiv	Hercules	CA		
Payankannur; Suresh	Richmond	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
IMX Mortgage Exchange	San Ramon	CA			02

APPL-NO: 08/ 928559 [PALM]
DATE FILED: September 12, 1997

INT-CL: [06] G06 F 17/00

US-CL-ISSUED: 705/38; 705/35, 705/37, 395/233
US-CL-CURRENT: 705/38; 705/35, 705/37

FIELD-OF-SEARCH: 705/38, 705/35, 705/37, 395/200.33, 395/200.47, 395/200.57, 395/200.61, 395/200.49

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
4750119	June 1988	Cohen et al.	
4799156	January 1989	Shavit et al.	
4876648	October 1989	Lloyd	
5136501	August 1992	Silverman et al.	
5297031	March 1994	Gutterman et al.	
5375055	December 1994	Togher et al.	

☐	5500793	March 1996	Deming et al.	
☐	5508913	April 1996	Yamamoto et al.	
☐	5560005	September 1996	Hoover et al.	
☐	5584025	December 1996	Keithley et al.	
☐	5592375	January 1997	Salmon et al.	
☐	5611052	March 1997	Dykstra et al.	705/38

OTHER PUBLICATIONS

XP002090668 Harker P.T. et al Aug. 1996.
XP002090669 Meyer D.L. et al Apr. 1992.
XP002090670 Colby M. Apr. 1993.
XP002090671 Reinbach A. Nov. 1996.

ART-UNIT: 277

PRIMARY-EXAMINER: Peeso; Thomas R.

ATTY-AGENT-FIRM: Swernofsky Law Group

ABSTRACT:

The invention provides a method and system for trading loans in real time by making loan applications, such as home mortgage loan applications, and placing them up for bid by a plurality of potential lenders. A transaction server maintains a database of pending loan applications and their statuses; each party to the loan can search and modify that database consistent with their role in the transaction, by requests to the server from a client device identified with their role. Brokers at a broker station can add loan applications, can review the status of loan applications entered by that broker, are notified of lender's bids on their loans, and can accept bids by lenders. Lenders at a lender station can search the database for particular desired types of loans, can sort selected loans by particular desired criteria, can bid on loan applications, and are notified when their bids are accepted. Broker stations, lender stations, and the transaction server can be coupled using multiple access methods, including internet, intranet, or dial-up or leased communication lines.

38 Claims, 2 Drawing figures

First Hit   Fwd Refs

Generate Collection   Print

L10: Entry 12 of 15   File: USPT   Nov 30, 1999

DOCUMENT-IDENTIFIER: US 5995947 A
TITLE: Interactive mortgage and loan information and real-time trading system

Abstract Text (1):
The invention provides a method and system for trading loans in real time by making loan applications, such as home mortgage loan applications, and placing them up for bid by a plurality of potential lenders. A transaction server maintains a database of pending loan applications and their statuses; each party to the loan can search and modify that database consistent with their role in the transaction, by requests to the server from a client device identified with their role. Brokers at a broker station can add loan applications, can review the status of loan applications entered by that broker, are notified of lender's bids on their loans, and can accept bids by lenders. Lenders at a lender station can search the database for particular desired types of loans, can sort selected loans by particular desired criteria, can bid on loan applications, and are notified when their bids are accepted. Broker stations, lender stations, and the transaction server can be coupled using multiple access methods, including internet, intranet, or dial-up or leased communication lines.

Brief Summary Text (11):
In a preferred embodiment, brokers at a broker station can add loan applications, can review the status of loan applications entered by that broker, are notified of lender's bids on their loans, and can accept bids by lenders. Lenders at a lender station can search the database for particular desired types of loans, can sort selected loans by particular desired criteria, can bid on loan applications, and are notified when their bids are accepted. Broker stations, lender stations, and the transaction server can be coupled using multiple access methods, including internet, intranet, or dial-up or leased communication lines.

Detailed Description Text (19):
The geographic database 112 includes a geocoded database accessed using a geographic database module. In a preferred embodiment, the geocoded database relates each property address to its corresponding census tract, and comprises information relating to each census tract, such as median income, used for computing possible CRA qualification for the loan. The transaction server 110 reads the geographic database 112 to determine information relating to the property.

Detailed Description Text (89):
The network 140 provides for communication between the transaction server 110, the broker station 120, the lender station 130, and the administration station 150, using messages as described herein and message protocols appropriate to transmission and reception of those messages. The network 140 includes a WAN (wide area network) such as the "Advantis" network available from IBM Corporation of Armonk, N.Y. However, in alternative embodiments, the network 140 may use internet, intranet, dial-up telephone lines or leased communication lines, or some combination thereof. In a preferred embodiment, the network 140 uses duplicate communication lines between nodes, and provides for automated failover transparent to users.

Detailed Description Text (93):

A method 200 of operation of the system 100 includes a plurality of flow points and process steps as described herein.

Detailed Description Text (94):
At a flow point 210, a prospective borrower desires to obtain a home mortgage loan or similar loan.

Detailed Description Text (116):
At a flow point 230, a prospective lender desires to bid on one or more loan applications.

Detailed Description Text (149):
At a flow point 260, a prospective borrower desires to accept a bid on a loan application.

Detailed Description Text (159):
At a flow point 280, a broker desires to prequalify a prospective borrower.

Detailed Description Text (162):
At a step 293, the broker at the broker station 120 enters preliminary information regarding the prospective borrower, such as--expected credit score, and expected front ratio, back ratio, and LTV ratio. The broker at the broker station 120 can enter information regarding points or rates, and obtain current and past corresponding points or rates; for example, the broker at the broker station 120 can enter a selected interest rate and obtain the average, high, and low values for points corresponding to that interest rate for completed loan transactions on the system 100. The broker at the broker station 120 can also determine statistics regarding how many lenders have made loans of those types.

First Hit Fwd Refs

End of Result Set

Generate Collection Print

L10: Entry 15 of 15 File: USPT Jul 2, 1996

US-PAT-NO: 5532838
DOCUMENT-IDENTIFIER: US 5532838 A

TITLE: Method & apparatus for dynamically creating and transmitting documents via facsimile equipment

DATE-ISSUED: July 2, 1996

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Barbari; Edward P.	Belmont	CA	94002	

APPL-NO: 08/ 422630 [PALM]
DATE FILED: April 14, 1995

PARENT-CASE:
This is a continuation of copending application(s) Ser. No. 08/175,212 filed on Dec. 27, 1993.

INT-CL: [06] H04 N 1/00

US-CL-ISSUED: 358/400; 358/403, 358/442, 358/444
US-CL-CURRENT: 358/400; 358/403, 358/442, 358/444

FIELD-OF-SEARCH: 358/400, 358/403, 358/407, 358/442, 358/434, 358/440, 358/444, 358/402, 358/468, 358/404, 340/995, 340/990, 340/998, 364/449, 364/443, 364/444, 364/436

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
4360875	November 1982	Behnke	364/436
4546439	October 1985	Esparza	364/444
4907274	March 1990	Nomura et al.	379/100
5197009	March 1993	Hoffman, Jr. et al.	364/443
5229947	July 1993	Ross et al.	364/443
5282052	January 1994	Johnson et al.	358/407

☐	5317628	May 1994	Misholi et al.	358/403
☐	5321520	June 1994	Inga et al.	358/403

OTHER PUBLICATIONS

Milwaukee, "Software Helps city Plan Trash Pickup", Government Computer News, V6, N18, p60(1), Sep. 11, 1987.
Map Info Corp., "W Marks the Spot-Map Info 2.0 for Windows", Data Based Advisor, V10, N12, p22(3), Dec., 1992.
A Status Report, "Front-Line Uses of GIS in Public and Private Sector Real Estate Today", Property Tax Journal, V12, N1, pp. 77-85, Mar. 1993.
Horizons Technology, "Horizons Releases SureMaps Version 2.0", Computer Reseller News, P. 172, Aug. 30, 1993.
MapInfo, "New MapInfo (TM) 4.0: Desktop Mapping on a Network", News Release, May 10, 1990.

ART-UNIT: 262

PRIMARY-EXAMINER: Coles, Sr.; Edward

ASSISTANT-EXAMINER: Lee; Fan

ATTY-AGENT-FIRM: Feix & Feix

ABSTRACT:

Computerized system for dynamically creating a document from a database responsive to telephonic input from a remote user, and transmitting the document to the user's facsimile machine. When a user desires to access a remote database according to the present invention, that user calls the apparatus of the present invention and in response to a voice menu system creates queries to the database. After receipt of the user's request, the system processes the request into a dynamically created document which meets the user's specifications and automatically transmits the document to the user's facsimile machine. A real estate sales support system constructed according to the principles of the present invention presents the user with a document comprising a listing of properties best meeting the user's needs and series of maps of varying scale enabling the user to locate the properties for inspection. Supporting the real estate sales support embodiment of the present invention is a methodology for geocoding listed properties with their respective latitudes and longitudes.

11 Claims, 40 Drawing figures

First Hit Fwd Refs
End of Result Set

Generate Collection Print

L10: Entry 15 of 15 File: USPT Jul 2, 1996

DOCUMENT-IDENTIFIER: US 5532838 A
TITLE: Method & apparatus for dynamically creating and transmitting documents via facsimile equipment

Abstract Text (1):
Computerized system for dynamically creating a document from a database responsive to telephonic input from a remote user, and transmitting the document to the user's facsimile machine. When a user desires to access a remote database according to the present invention, that user calls the apparatus of the present invention and in response to a voice menu system creates queries to the database. After receipt of the user's request, the system processes the request into a dynamically created document which meets the user's specifications and automatically transmits the document to the user's facsimile machine. A real estate sales support system constructed according to the principles of the present invention presents the user with a document comprising a listing of properties best meeting the user's needs and series of maps of varying scale enabling the user to locate the properties for inspection. Supporting the real estate sales support embodiment of the present invention is a methodology for geocoding listed properties with their respective latitudes and longitudes.

Brief Summary Text (7):
Typically, automated map systems will provide a map taken from a database of maps for a given location. To match the requested location with the correct map, or portion of a map, some kind of geocoding system is used, most often based on latitude and longitude. Again, users of automated map generation systems require knowledge of the system, a computer and, if printed output is desired, a connected printer. To produce a map for a given location, the location's latitude and longitude (or other geocoding input) are entered, and a map produced for that location. A system which would input a list of locations and extract from the map system a map or series of maps for these locations would provide real estate sellers with a powerful sales tool at minimal expense of effort. To be truly useful, however, maps provided by the system should have several characteristics not found in the map systems currently available.

Brief Summary Text (9):
A common fault of automated map systems currently available is that a map produced therewith for a given location often depicts that location at the map boundary. A map with a listed property shown at one edge or in a corner of the map has limited utility. Maps presented to the user which "center" the location of the property or other point of interest on the map produced would maximize the utility of those maps. Furthermore, assuming the user is provided with a series of maps of varying scale as discussed above, the location of the detail maps should be centered on the area maps, and the location of the area maps should be centered on the general maps. In this manner, the user is given the most useful presentation of map data.

Brief Summary Text (10):
To maintain a dynamic database of listed real estate properties, properties need to be removed when sold or otherwise removed from the market and added when they come

on the market. In order to create a map which will accurately depict newly listed properties, each property in the database must contain geocode references for a mapping system to refer to. Manually geocoding each new property as it is listed would be a significant effort; absent a large expenditure of man-hours it cannot be timely done for the large pool of available properties typically found in even medium sized cities. An automated geocoding process which takes advantage of geocode references inherent in an available mapping system would solve the problem, if those geocode references are accurate and up to date. Geocode references in available automated mapping systems are constantly changing to reflect changes to the areas they map; furthermore, these systems almost always have inherent errors or inaccuracies. A purely automated process to provide geocode references for a large number of newly listed properties is not, therefore, predestined for success. An automated process, capable of manual intervention to resolve errors, would enable a database manager to geocode the greatest number of properties by maximizing the utility of the geocoding capability of existing mapping systems, while minimizing the impact of those errors.

Brief Summary Text (13):
Using a facsimile machine to receive printed output from remote storage locations has been the subject of efforts by several workers. U.S. Pat. No. 4,918,722 to Duehren et al. teaches a method for storing at one location messages, receiving at the first location commands via telephone lines from a second location which select one of the stored messages, fetching the selected message responsive to the commands, calling the second location and sending the selected message to that location via facsimile equipment.

Brief Summary Text (20):
To practice the present invention, a database containing real estate offerings is implemented on a computer using a commercially available database management system. The present invention provides means to process the database, geocoding the real estate listings with their respective latitude and longitude to a pre-determined degree of accuracy. When a given listing cannot be automatically geocoded, the present invention provides means for the system operator to manually intervene in the geocoding process.

Brief Summary Text (21):
Once the database is geocoded, the present invention enables a remote user or subscriber (hereinafter the user) to access a computerized database system via DTMF telephone, and to create from that database system a document in response to the user's specific needs. In the real estate sales support embodiment of the present invention, the resultant document comprises a listings page with a selection of real estate properties best meeting the user's needs as input, and a corresponding series of real estate locator maps. The maps produced by this embodiment of the present invention include one or more detail, area, and general maps. The detail map or maps show the detailed location for each of the property listings; the area map or maps show the location of each of the detail maps on a smaller scale map; and the general map or maps show the locations of the area maps on a very small scale map.

Brief Summary Text (23):
The user's request is input via DTMF telephone codes in response to voice prompts from the present invention. Among other information, the voice prompts of the present invention ask the user to input the area the user is interested in, the target price the user desires to pay for a property in this area, and a valid fax number where the resultant document will be sent. The present invention then processes the user's request, creates a customized document in response to that request, and transmits the document via telephone lines to a facsimile machine to which the user has access. The voice prompts are created by a voice processing card, installed in a computer, the card being responsive to a voice processing control software loop. Voice processing cards are well known in the art, but a

particular feature of the present invention is the methodology used therein to interactively couple the voice prompts with the user's responses to query the several databases of the system.

Brief Summary Text (25):
Once the database containing the real estate listings has been geocoded, the present invention is designed to operate essentially continuously with minimal system operator input. The present invention iteratively and continually tests incoming phone lines for incoming calls and automatically processes each call. As a user calls the system, he or she is greeted by a voice prompt, and asked to input the area desired. This is done by using the letter codes on a DTMF telephone. Testing has determined that approximately 80% of all areas can be accurately input using the first five letters of an area's name. Accordingly, the present invention requires only those first five letters, in order to save both user and processing time.

Brief Summary Text (29):
Individual document pages created by the present invention in response to a given user request are stored as separate files. All related files representing a single request document are stored until transmitted by the fax transmission means of the present invention. The present invention provides for multiple attempts at facsimile transmission to minimize the impact of busy lines at the facsimile receiving number specified by the user. Once documents are transmitted, the present invention provides means for deletion of their corresponding files to save computer storage space. Means are likewise provided by the present invention for several record keeping functions of import to the system operator. These functions are discussed in detail in the section entitled "BEST MODE OF CARRYING OUT THE INVENTION".

Drawing Description Text (28):
FIG. 26 is a detail flow diagram of the center map data--update ML2 database with area map center point/create individual request map data file module of the process new request module.

Drawing Description Text (40):
FIG. 38 is an overview of the geocoding process of the present invention.

Detailed Description Text (3):
To practice the present invention, a database containing real estate offerings is implemented on a computer using a commercially available database management system. Each real estate offering is accompanied in its database record with a unique positional locator comprising its latitude and longitude. The association between a listed property and its latitude and longitude is accomplished through a process called geocoding. The present invention is practicable using a wide variety of geocoding methodologies. The simplest method would be to have a subscriber who desired to have his property listed with a system operator using the present invention to provide the positional locating data for his property, as latitude and longitude, to the system operator when subscribing. Another method could be for a subscriber to furnish this data through a second source provider, i.e., another firm providing automated mapping capabilities. Still another methodology would be a combined manual and automated process practiced by the system operator in conjunction with the present invention. In this scenario, the system operator, through a combination of manual and automated means processes the database, geocoding the real estate listings with their respective latitude and longitude to a pre-determined degree of accuracy. When a given listing cannot be automatically geocoded, the present invention provides means for the system operator to manually intervene in the geocoding process.

Detailed Description Text (4):
Geocoding is the process by which mapping coordinates in terms of longitude and

latitude can be associated with or determined for a given location. Most proprietary mapping databases have some geocoding capability. In general, on one extreme, the geocoding process can be highly interactive requiring considerable human involvement and decision making on a location-by-location basis. On the other extreme, the geocoding process can be highly automated requiring pre-defined search strategies applied to large numbers of locations. In both cases, the geocoding is a reiterative process in which multiple degrees of accuracy are applied during each geocoding attempt. In the present invention, it is necessary to geocode a large number, often several thousand, locations in a rather short period of time. Ideally, the process should be capable of accomplishment in one to two hours. To accomplish such a task, the geocoding process must be automated as much as possible.

Detailed Description Text (5):
In order to plot the position of a given location on a map, its mapping coordinates must be known. In an application providing locator maps for groups of locations, this information must be determined in advance. Source information in terms of address (the numeric portion of a street address), street, city, state, and ZIP code must be processed in a systematic manner under a predefined geocoding strategy. The geocoding strategy applied must be designed to take into consideration the completeness and accuracy of the source information, the completeness and accuracy of the mapping database geocoding index files, and the processing capabilities available to address weaknesses in both sources of information.

Detailed Description Text (6):
During the process, success for a given attempt is measured by a single set of coordinates being returned which are within a pre-defined range of positional accuracy. Non-successful attempts can be classified into two groups. One group is characterized as those which must be investigated or processed manually to determine their viability for geocoding. The other group is characterized as those which can continue to be processed automatically under less restrictive parameters for positional accuracy.

Detailed Description Text (7):
The success rate for a totally automated geocoding process will vary according to the accuracy of the address information submitted and the accuracy and completeness of the mapping database geocoding index files. Given the potential for errors in both sets of information and the difficulty of maintaining geocoding index files for the latest information on new streets and addresses, a totally automated geocoding process cannot be expected to be successful on all attempts.

Detailed Description Text (8):
For the percentage that cannot be geocoded within a pre-defined range of positional accuracy, the objective is to then geocode to within a pre-defined area. This geocoding process can also be highly automated.

Detailed Description Text (9):
For the percentage that cannot be geocoded either within a predefined range of positional accuracy or within a pre-defined area, the geocoding process is a manual effort. It is quite possible that a very small percentage of locations will not be able to be geocoded without a substantial manual effort.

Detailed Description Text (10):
Since geocoding strategies are highly dependent on the capabilities of the geocoding tools and the geocoding index files associated with a given mapping database, the following discussion will focus on specifics pertaining to the mapping products developed by ETAK, Inc.

Detailed Description Text (11):

Geocoding information from ETAK is encoded in a proprietary format in a GeoIndex file. The information contained in this file consists of street name, city, state, postal code (ZIP code in the United States), address range, and coordinate range. According to ETAK, geocoding is the process of fetching information about a location on a map from a given description of the location. Its primary goal is to provide a set of geographical coordinates in response to an input address specification.

Detailed Description Text (12):
The input address specification can consist of six possible components: 1) an address number, 2) a street name, 3) a cross street name, 4) a city name, 5) a state name, and 6) a postal code. Because cross street names are generally not available and because the geocoding results from their use does not provide the positional accuracy required by the present invention, they are not used in the geocoding strategy outlined below. The street name as well as the cross street name components can be broken into three smaller components: a prefix, such as a directional modifier--north, south, east, west, etc.; a suffix such as a street type--drive, avenue, boulevard, court, etc.; and a body. The body is the part of the street name that is left after stripping the prefix and suffix. The street name body is the only required component for a geocoding search since it is present in every GeoIndex entry and must be present in any valid street address.

Detailed Description Text (13):
When submitting an input address specification for geocoding using ETAK's GeoIndex file, the street body is required. All other address components, including address number, prefix, suffix, city, state, and postal code are optional. ETAK provides the capability to set search flags on each of these components. These flags can either require an exact match or allow a non-match, or wild card.

Detailed Description Text (14):
Since there are a large number of permutations of components and search flags using ETAK's GeoIndex file, an optimized automated geocoding strategy must be designed to maximize the success rate with the minimum number of iterations. In processing a large number of input address specifications, the iterative processing to execute a geocoding strategy can be applied in its entirety to each record or as separate iterations applied to the entire file of records. The geocoding--processing overview in FIG. 38 outlines the processing for a strategy applied as separate iterations to an entire file of records.

Detailed Description Text (15):
The geocoding process starts as a manual process. Certain errors in the data are corrected prior to submission to the automated process. At the beginning of the automated process, the criteria used to initiate the searching process is set. This includes the various search flags on each of the components. Each of the components are then tested for a valid entry in the GeoIndex file. If any of the components of an address are not valid, then the address is considered to contain bad components and cannot be submitted for geocoding. These bad components must be manually examined to determine the source of the problem. In the initial stage of applying a geocoding strategy, the source of the problem is usually the data itself. Errors are corrected manually and the corrected information is resubmitted to the automated geocoding process as soon as possible at the test components step. In the later stage of applying a geocoding strategy, the source of the problem is usually the GeoIndex file where information may be either incorrect or missing. After identification of GeoIndex file errors and possibly omissions, the data can be resubmitted to the automated geocoding process where it is adjusted prior to the test components step.

Detailed Description Text (16):
If all of the components are valid, the address is geocoded. If there is a single match, a determination is made as to whether the criteria being applied provide for

acceptable positional accuracy. If the accuracy is acceptable, this is considered a successful geocode and the coordinates from the process are stored for that location. If there are multiple matches, a determination is made as to how these will be processed. If manual processing is appropriate, these will be reviewed manually and the correct coordinates will be stored.

Detailed Description Text (17):
If there is a non-match, or a single match but with unacceptable positional accuracy, or multiple matches which can be processed in an automated manner, a determination is made relative the geocoding strategy. If the strategy has not been exhausted in terms of options as designed, the search criteria is changed and the records are resubmitted at the test components step. If the strategy has been totally applied, the unsuccessful attempts will be processed manually.

Detailed Description Text (18):
The specific geocoding strategy used in conjunction with the present invention makes several passes with specific objectives in mind. On the first pass, the objective is to successfully geocode as many locations as possible with the highest degree of positional accuracy possible working from the tightest criteria and gradually loosening the search criteria. On the second pass involving only the unsuccessful records from the first pass, the objective is to separate the data into two groups. One of these groups represents data which cannot be geocoded in an automated manner using ETAK's Geolndex file and will ultimately be geocoded by some other means. The other group represents data which may be geocoded within some range of positional accuracy. On the third pass involving only this last group, the objective is to successfully geocode as many locations as possible within the pre-defined limits of positional accuracy again working from the tightest criteria and gradually loosening the search criteria. After these three passes, the remaining unsuccessful matches are adjusted for errors and/or omissions known to exist in the ETAK Geolndex file. At this point, another three passes with the same objectives as the first three are applied to the data. After this process, the remaining unsuccessful matches are submitted to a geocoding strategy based upon ZIP code information contained in a non-ETAK file. By the end of this process, the number of locations requiring manual processing is minimal.

Detailed Description Text (32):
A particular feature of the real estate sales support embodiment of the present invention is the manner whereby listings are centered on the maps presented to the user as part of the document which is ultimately delivered. Prior to processing user requests, the system operator creates the location database which defines the several areas/neighborhoods requestable by users when they access the system. Open home listings are linked to the various areas/neighborhoods defined in the location database. Referring to FIG. 39, each area/neighborhood is defined by latitude and longitude coordinates representing its center point, and an area size in miles which represents one-half of the width and height of the area. To eliminate the possibility of detail map boxes drifting off the page, a buffer of 0.4 miles on all sides is used to reduce the area size for purposes of calculating the parameters for linking open home listings for any given area/neighborhood.

Detailed Description Text (34):
Responsive to processing a user's request, open home listings meeting a user's specifications are linked to detail maps. As the request is processed, the latitude and longitude of the first selected property listing becomes the center point of a detail map. When an additional property listing will fit on an existing detail map, coordinates representing the detail map's center point are recalculated based on the midpoints of the latitudinal and longitudinal extremes of the selected locations sited on each detail map. Each detail map therefore is dynamically changed pursuant to processing the user's request. Referring to FIG. 40, each detail map is defined by its centerpoint which is based on the midpoints of the latitudinal and longitudinal extremes of the selected locations and an area size in

miles which represents one-half of the width and height of the detail map area. The default area size for detail maps is set at 0.4 miles for a 0.64 square mile area (0.8 * 0.8). To eliminate the possibility of open home listings appearing too near the edge, a buffer of 0.1 miles on all sides is used to reduce the area size for purposes of calculating the parameters for linking open home listings to any given detail map.

Detailed Description Text (35):
Just as detail map center coordinates are dynamically modified to center the detail map around adjacent property listings, so are area maps dynamically modified to center the area map around the midpoints of the several detail maps produced responsive to a user's request. Referring to FIG. 40, as detail map center coordinates are linked to the area map, coordinates representing the area map's center point are recalculated in the same manner. The area map is therefore defined by these coordinates and its area size in miles, which represents one-half of the width and height of the area map area.

Detailed Description Text (38):
This embodiment presents three levels of mapping. Specifically, detail maps, area maps and general maps. Detail maps are large scale maps in that they portray a small area at great detail. Area maps are medium scale maps and general maps are small scale maps in that they portray a large area. In the embodiment of the present invention under discussion, the user is presented with one or more detail maps. Each detail map has imprinted thereon one or more numbered dots, also referred to herein as map points. Each numbered dot corresponds with a located property as well as the property listings shown on the listings page of the document prepared by the present invention. It is a specific feature of the present invention that these numbered dots are centered with respect to the detail maps. In the event where a detail map has only one numbered dot, the present invention provides means whereby that dot appears at the center of the detail map. In the event a detail map has two or more numbered dots corresponding to property locations, the center of the detailed map is based on the midpoints between the longitudinal extremes and latitudinal extremes of all the data points associated with that map. In similar fashion, the location of the detail maps on the area map is centered using the same methodology. Furthermore, the location of the area map or maps on the general map is likewise centered.

Detailed Description Text (39):
The principal databases used in centering map data are as follows: Map Level 3, or ML3 is a database which has records stored therein pertaining to each individual listing point selected by the system as most closely approximating the user's requirements. The records in Map Level 2, or ML2 define the center points of each detail map. The center coordinates of the detail maps will become the locational point on the area map. Each record in ML2 will result in one detail map created by the system. Map Level 1 or ML1 will result in the center point or locational point for the area and general maps. The data in each of these databases are dynamically created by the system responsive to each user's input, and are based on property listings previously stored in the listings database. ML3 will typically have several entries contained therein for each user request. ML2 will have one or more entries, each of which contains the center point for one detail map. As ML2 database is created, the centerpoint for any map represented by one of its records may change, as explained below. ML1 will contain, after the map centering process, a number of records equal to the number of areas the system will allow the user to input in the user's request.

Detailed Description Text (40):
Map centering is effected as follows: the ML3 database is created from the individual request database and updated to include information pertaining to the location of the several listings selected for inclusion by the system in the document. Once the ML3 database has been updated, it is indexed by its latitude and

longitudinal coordinates from north to south and west to east. Before the system opens and initializes the ML2 database, the system then defines the map size, border buffer and calculates positional boundaries. Positional boundaries are the map size dimension less the border buffer dimension--for both latitude and longitude. The positional boundaries represent the maximum distance from the center point of a map that a point can be and still be associated with that particular map.

Detailed Description Text (43):
Another feature particularly pointed out by the present invention is the use of border buffers. Border buffers are minimum areas around the perimeter of a map which, while they contain all the detail of the map, will contain no location points. This is to ensure that no map is produced with a position location at the edge of the map, and hence possibly lacking sufficient locational detail as to allow a user to find the property.

Detailed Description Text (44):
The first record in ML3 causes the system to create the first record in ML2. This is equivalent to a first detail map being created to display a first location. Each subsequent record in the ML3 database is then sequentially processed, and checked to see if it will fit on the map corresponding to the current ML2 record. If the point will fit, without exceeding the positional boundaries for the detail map, it is added to that map, and the ML3 database map number is updated from the ML2 database. The internal windows are re-adjusted and the map center coordinates are re-calculated. This information is updated to ML2 so when the next set of tests are conducted, the positional boundaries of the map in the ML2 database record are re-determined. If it will not fit, the system tries to fit the point, in similar fashion, to each of the previously recorded ML2 records. If a fit is found, the system adds the point to the map which fits the point. If no such fit is found, the system creates a new ML2 record, and hence a new detail map before checking the next ML3 record.

Detailed Description Text (45):
When the first point is put in an ML2 record, that record defines a detail map centered around that first point. When a second or subsequent point is put on the same detail map, the system recalculates a new map center based on the midpoints between the longitudinal extremes and latitudinal extremes of all the data points associated with that map. This new map center is stored in the ML2 database to form the center of the detail map as it is eventually drawn.

Detailed Description Text (46):
At this point in executing the program of the present invention, the ML2 database is created, and each of the detail maps defined thereby has had its center point calculated. The ML3 database has been updated to include corresponding map numbers on the ML2 database. The present invention now indexes both databases by map number, links ML3 to ML2 by map number and ensures that the detail map center coordinates from ML2 are updated on ML3. The ML1 database is now opened and initialized, the map size and border buffer defined, and the positional boundaries calculated for the area map.

Detailed Description Text (47):
The first record in ML2 causes the system to create the first record in ML1. This is equivalent to a first area map being created to display a first detail map. Each subsequent record in the ML2 database is then sequentially processed, and checked to see if it will fit on the map corresponding to the current ML1 record. If the point will fit, without exceeding the positional boundaries for the area map, it is added to that map, and the ML2 database map number is updated from the ML1 database. The internal windows are re-adjusted and the area map center coordinates are re-calculated. This information is updated to ML1 so when the next set of tests are conducted, the positional boundaries of the map in the ML1 database record are

re-determined. If it will not fit, the system tries to fit the point, in similar fashion, to each of the previously recorded ML1 records. If a fit is found, the system adds the center point of the detail map to the area map which fits the center point. If no such fit is found, the system creates a new ML1 record, and hence a new area map. When the first detail map center point is put in an ML1 record, that record defines an area map centered around that first detail map center point. When a second or subsequent detail map center point is put on the same area map, the system recalculates a new map center based on the midpoints between the longitudinal extremes and latitudinal extremes of all the data points associated with that map. This new map center is stored in the ML1 database to form the center of both the area and general maps as they are eventually drawn. If the embodiment of the present invention allows several areas to be selected by the user, the number of records in the ML1 database will equal the number of areas the user selects. If the present invention is configured to allow only one user selected area, there will be only one record in the ML1 database.

Detailed Description Text (49):
At this point the map data database is opened and initialized, and records from the ML3, ML2 and ML1 databases are merged into it. Data is then copied from the map data database into the individual request map data file, and the map data database is closed.

Detailed Description Text (52):
Referring now to FIG. 1, two computers, 1 and 2, are shown configured as a local area network. Well known is the fact that such computers each typically comprise a CPU, input and output means, bus means, display means, storage and memory means. In the best mode for carrying out the invention, each of computers 1 and 2 is equipped with dual input means specifically both a keyboard 3 and 4 and a mouse 5 and 6. In order to connect computers 1 and 2 to form a local area network for carrying out the present invention, they are connected by cable 10 to network interface cards 14 and 15 (shown in FIG. 2), installed in computers 1 and 2 respectively. It will be obvious to those skilled in the art that this is not the only methodology whereby the present invention may be carried out. A single computer capable of multiprocessing or a computer with one or more attached co-processors could be configured to carry out the present invention in a single computer configuration. In addition to the previously discussed general architecture requirements for a general purpose, digital programmable computer, computer 1 in the present invention has installed therein a voice processing card 13 capable of receiving one or more incoming telephone lines. Computer 1 has further implemented thereon one or more fax modem card 60 capable of transmitting facsimile copies generated by the system. Fax modem cards are in turn connectable to one or more outgoing telephone lines.

Detailed Description Text (54):
The operating system used in perfecting the real estate sales support system embodiment of the present invention is Microsoft MS-DOS version 5.0. C language programming was carried out using Borland C++ version 3.1. The database processing is conducted using Borland's dbase IV.TM. version 2.0. Voice processing software from Telephone Response Technology comprises 3 modules: ProVIDE Application Processor version 4.2c; DBA module version 4.1b; and INTELEFAX module version 1.7e. Map drawing routines utilize MapAccess.TM. tools from ETAK: MapDraw library version 2.0.2 and MapRetreive library version 1.0.2. MapAccess.TM. GeoCode library version 1.4.1 provides the interface for the automated geocoding process of the present invention. The conversion from PostScript.TM. format to PCX is conducted by GS32 from LaserGo.

Detailed Description Text (55):
An overall system view of the present invention is shown in FIG. 2. Incoming analog phone line 12 connects to voice processing card 13. Voice processing card 13 operates on computer 1 which executes voice processing control loop 100. Voice processing control loop handles incoming telephone calls during which a user is

interactively prompted to make a request for information via touch tone responses. Each request requires the user to select an area, enter a target price, and provide a telephone number to which the requested information will be sent.

Detailed Description Text (56):
Also resident on each of computer 1 and computer 2 is a network interface card, 14 and 15 respectively, connected by cable 10 to configure the computers as a peer-to-peer local area network, shown in FIG. 2. Voice processing card 13 and fax card 60 serve to connect the present invention to incoming and outgoing telephone lines 12 and 70, respectively. Requests for information generated by the voice processing control loop are forwarded to the database system resident on computer 2 via the local area network. The request processing and supervisory control loop 200 resident on computer 2 processes all new requests, monitors fax transmissions of requests, and automatically deletes fax files at a predetermined time.

Detailed Description Text (57):
Requests which have been processed and the documents created responsive to those requests are sent from computer 2 to computer 1 again by the peer-to-peer local area network. Facsimile transmission of the maps and listings sheets produced by the system is controlled by fax processing control loop 300. Fax processing control loop 300 executes and monitors transmission of the finalized map set from fax card 60 via outgoing analog telephone line 70.

Detailed Description Text (58):
Voice processing control loop 100 is detailed in FIG. 3. The present invention is designed to operate essentially continuously with minimal system operator input. The present invention iteratively and continually tests incoming phone lines for incoming calls and automatically processes each call. At 101 the loop tests voice processing card 13 for an incoming call. If no incoming call is detected, the test is repeated. If an incoming call is detected, the call is processed by incoming call module 105, which comprises main service menu and open home menu 110, area selection module 130, target price input module 170, and fax number input and validation module 185. After an incoming call is processed by process incoming call module 105, the request queue database is updated at 103. After the request queue has been updated, voice back closing 104 signs the system off to the user, closes the phone line and returns control to test 101, which iteratively continues to test for incoming telephone calls as long as the system is operating.

Detailed Description Text (60):
The present invention provides for multiple attempts at facsimile transmission to minimize the impact of busy lines at the facsimile receiving number specified by the user. Fax processing control loop 300 introduced in FIG. 2 is outlined in FIG. 5 as follows. After a pause in processing at 301, the fax processing control loop opens the fax queue database and reads the fax queue database status code at 302. A test is made for new work in the queue at 303. If there is no new work in the queue, a return is made to pause 301, which in the preferred embodiment of the present invention, comprises six seconds. If there is new work in the queue, i.e., the status code has been set to zero, process outgoing call module 304 is invoked. Outgoing fax call module 304 contains two main modules, document queue and transmission module 320 and fax call monitoring module 360. After the sequential invocation of these two modules, process outgoing fax call module 304 invokes update fax queue database module 305. After step 305 updates the fax queue database, the control loop returns to pause processing step 301, previously discussed.

Detailed Description Text (74):
Fax call monitoring module 360 is detailed in FIG. 15. At step 361, the call retry switch, fax duration, and completion status variables are initialized and reset. A pause is made at 362 for call initiation. At 363 the line status is obtained, and then tested at 364. If the line status test indicates "in process", the call retry

switch is reset at 365, and the line test is paused for 10 seconds at 366 before looping back to get line status at step 363. If line status test 364 indicates idle, the call retry switch is tested for previous retries of the line idle test at 367. If no previous retries are encountered, the call retry switch is set to 1 at step 368 before looping back to get line status at step 363. If test step 367 determines the previous retries switch to be set to 1, a log retry switch is initialized and reset at step 369. After initializing step 369 is conducted, a pause is made for 15 seconds at step 370 before testing the log file for fax results. Following the pause at 370, the fax duration and completion status are obtained from the log file at 371, and a test made at 372 to test the fax duration and completion status. If the fax duration is zero and the completion status is zero, indicating the log file has not been updated, a loop returns execution to step 370. If the fax duration is zero and the completion status is not zero, or if the fax duration is not equal to zero and the completion status is equal to zero, (indicating a successful send) the log retry switch is tested at 373. If no previous retries are encountered, the log retry switch is set to 1 at 374 and a loop returns execution to step 370. If the fax duration is not equal to zero and the completion status is not equal to zero, or if log retry switch test step 373 determines the previous retries switch to be set to 1, the update of the fax queue database step at 305 previously discussed in FIG. 5 is invoked. Referring back to line status test step 364, if a fax error code 7 or fax aborted code 5 is detected, execution passes to step 371, get fax duration and completion status.

Detailed Description Text (75):
The new request process module 400, introduced in the new request test loop as outlined in FIG. 11, is outlined in FIG. 16. Process new request module 400 is comprised of six sequentially invoked modules: Update request files 410; extract listings module 490; create listings page module 800; center map data module 900; create map pages module 1250; and module 1720 which performs several files management functions. Update request files 410 is invoked by step 221 preceding. Update request files 410 invokes extract listings module 490. Extract listings 490 is further comprised of three sequentially invoked modules: initialize parameters module 500, which also processes requests consisting of four listings or less; processing for more than four listings module 600; and extract and update databases module 700. After execution of module 490, execution passes to create listings page module 800, which in turn invokes center map data module 900. Center map data 900 is comprised of five subordinate modules. Module 901 initializes and creates map level 3 database and initializes map level 2 database. Module 970 creates map level 2 database, which contains the detail map center points. Module 1030 updates map level 3 database with detail map center points and initializes map level 1 database. Module 1050 creates map level 1 database, which contains the area map center points. Module 1160 updates map level 2 database with the area map center point and creates the individual request map data file. Sequentially invoked after center map data module 900 is create map pages module 1250, comprised of three subordinate modules: initialize files/set up functions module 1260; main processing loop 1290; and complete and close files module 1700. After the map pages are created, module 1250 invokes module 1720, which converts PostScript.TM. files to PCX format, renames and copies PCX files, deletes intermediary working files, and updates fax queue and request databases. The several component modules of the process new request module are discussed in detail as follows:

Detailed Description Text (81):
After creating a listings page at module 800, the system of the present invention centers the map data on each of the maps created in the following manner: The numbered dots 75 are centered on the detail maps 80 shown on FIGS. 34 and 35. The detail maps 80 are centered on the area map 85 and the area map 85 is centered on the general map 90. Where more than one numbered dot or map outline is located on a given map, the present invention recalculates a new map center for that map each time a new numbered dot or map outline is added to the map. The new map center is based on the midpoints between the longitudinal extremes and latitudinal extremes

of all the data points associated with that map. For a map outline, the relevant data point is the map center point for that outline. After a new map center is calculated, it is used to determine if any new dots or outlines can be added to that particular map. If a new data point cannot fit within the boundaries of any map previously defined during the map centering process, a new map is created to accommodate that data point. This new map plus those previously defined during the map centering process are then used for positioning any subsequent additional data points. Center map data module 900 is comprised of five main modules: Initialize and create map level 3 database and initialize map level 2 database module 901, create map level 2 database module 970, updata map level 3 database with detail map center points and initialize map level 1 database module 1030, create map level 1 database module 1050, and update map level 2 database with area map center point and create individual request map data file module 1160. Each of these modules will be discussed below.

Detailed Description Text (82):
Center map data module 900 is accessed at module 901, which initializes and creates the map level 3 database and initializes the map level 2 database. Module 901 is detailed at FIG. 22. Module 901 is initiated at step 902 which opens the request database, locates the request number and stores the corresponding location file record number, and then closes the request database. Execution then passes to step 910 which opens the location database, locates the location file record number and stores the area name and area size before closing the location file. At step 920, map level 3 database is created from the individual request database. At step 930, the map level 3 database is updated to include the map level, the map number, the coordinates as latitude and longitude, location dot number, description line 1 which includes address, street and unit, and description line 2 which includes the date and time the listed property is open for viewing, the map scale, and the box scale. At step 940, the coordinates of each of the listed properties are sorted from north to south and west to east. At step 950, map level 2 database is opened and initialized. At step 960 the map size and border buffer are defined and the position of boundaries is calculated. Positional boundaries are the map size dimension less the border buffer dimension--for both latitude and longitude. The positional boundaries represent the maximum distance from the center point of a map that a given point can be and still be associated with that particular map. Step 965 moves to the beginning of the map level 3 database, before passing control to module 970 at step 971.

Detailed Description Text (83):
Module 970 detailed at FIG. 23 creates the map level 2 database containing the detail map center points. The module is invoked by step 965 of the preceding module. At step 971, a test is made of the map level 3 database to determine that the current record is not past the end of the file. If the current record is not past end of file, map level 2 database is accessed at the beginning of the database at step 972. At step 973 a test is made for records in the map level 2 database. If at least one record exists, a test is made at step 974 that the current record is not past the end of the file, again referring to map level 2 database. If the record is not past the end of the file, the current map number is stored as a variable at step 975. Step 976 calculates the latitudinal and longitudinal distance of the individual listing point from the current map center. At step 977, a determination is made if the current point can be added to the internal window without exceeding the positional boundaries for the current map. If the current point does not fit within the positional boundaries, execution passes to step 978 which moves to the next record in the map level 2 database before looping back to step 974. If the determination at step 977 finds that the current point will fit within the internal window, four actions are taken: Map level 3 database is updated with the map number from map level 2 at step 979. The internal window coordinates are adjusted at step 980 and a new map center recalculated at step 990. At step 1000, map level 2 database is updated to include latitude and longitude coordinates as well as the internal window coordinates. After step 1000 executes, step 1010 is

invoked, where a test is made of the record in the map level 3 database for an updated map number. Referring back to step 974 where map level 2 database is tested for being not past the end of the file, if the test determines that the database is past the end of the file, step 1010 is likewise invoked. If test step 1010 determines that the record in the map level 3 database does not have an updated map number, step 1011 adds a record to the map level 2 database and updates the map number by incrementing the current map number by 1, updates location coordinates with latitude and longitude from the map level 3 database, and updates internal window coordinates. Step 1011 then updates the map level 3 database with the map number from the map level 2 database before passing to step 1025. Step 1025 moves to the next record in the map level 3 database before executing a loop back to step 971. Referring back to step 973, if the test for no records determines that no records are present in the map level 2 database, step 1020 is invoked, which adds a record to the map level 2 database and updates the map number, updates location coordinates with latitude and longitude from the map level 3 database, and updates internal window coordinates. Step 1020 then updates the map level 3 database to include the map number before passing to step 1025. Referring back to test step 1010, if the test step determines that the map level 3 database record has an updated map number, control is directly passed to step 1025 without invoking either step 1011 or step 1020. Referring back to step 971, once the test of the map level 3 database determines that the record is past the end of the file, module 970 is terminated and module 1030 is invoked.

Detailed Description Text (84):
Module 1030, detailed at FIG. 24, updates map level 3 database with detail map center points and initializes the map level 1 database. Module 1030 is accessed from step 971 of module 970 at step 1031. Step 1031 indexes the map level 2 database by map number, indexes the map level 3 database by map number, and links the map level 3 database to the map level 2 database on the map number before passing to step 1034. Step 1034 moves to the beginning of the map level 3 database. At step 1035, a test is made of the map level 3 database to determine that the current record is not past the end of file. If it is not past the end of the file, at step 1036 map level 3 database is updated to include the map center coordinates from map level 2 before the present invention accesses the next record in the map level 3 database at step 1037. Step 1037 loops execution of the program back to step 1035. If step 1035 determines that the record in map level 3 database is past the end of the file, step 1038 closes the map level 3 database and step 1039 opens the map level 2 database. Step 1040 updates the map level 2 database with the map number and map level before passing control to step 1041 which indexes coordinates from north to south and west to east. At step 1042, map level 1 database is opened and initialized. Step 1043 defines the map size, border buffer, and calculates positional boundaries of a new record in the map level 1 database before moving to step 1044. Step 1044 moves to the beginning of the map level 2 database and invokes module 1050.

Detailed Description Text (85):
Module 1050, detailed at FIG. 25, creates the map level 1 database containing the area map center point. The module is invoked at step 1056 by step 1044 of the preceding module. At step 1056, a test is made of the map level 2 database to determine that the current record is not past the end of the file. If the current record is not past end of file, map level 1 database is accessed at the beginning of the database at step 1057. At step 1058 a test is made for records in the map level 1 database. If at least one record exists, a test is made at step 1059 that the current record is not past end of file, again referring to map level 1 database. If the record is not past the end of the file, the current map number is stored as a variable at step 1060. Step 1060 also calculates the latitudinal and longitudinal distance of the detail map center point from the current map center. At step 1080, a determination is made if the current point can be added to the internal window without exceeding the positional boundaries for the current map. If it does not fit within the positional boundaries, execution passes to step 1100

which moves to the next record in the map level 1 database before looping back to step 1059. If the determination at step 1080 finds that the current point will fit within the internal window, four actions are taken: Map level 2 database is updated with the map number from map level 1 at step 1101. The internal window coordinates are adjusted at step 1102 and a new map center recalculated at step 1110. At step 1120, map level 1 database is updated to include latitude and longitude coordinates as well as the internal window coordinates. After step 1120 executes, step 1130 is invoked, where a test is made of the record in the map level 2 database for an updated map number. Referring back to step 1059 where map level 1 database is tested for being not past the end of the file, if the test determines that the database is past the end of the file, step 1130 is likewise invoked. If test step 1130 determines that the record in the map level 2 database does not have an updated map number, step 1135 adds a record to the map level 1 database and updates the map number by incrementing the current map number by 1, updates the location coordinates with the latitude and longitude from the map level 2 database, and updates the internal window coordinates. Step 1135 then updates the map level 2 database with the map number from the map level 1 database before passing to step 1155. Step 1155 moves to the next record in the map level 2 database before executing a loop back to step 1056. Referring back to test step 1058, if the test for no records determines that no records are present in the map level 1 database, step 1145 is invoked, which adds a record to the map level 1 database and updates the map number, updates the location coordinates with latitude and longitude from the map level 2 database, and updates internal window coordinates. Step 1145 then updates the map level 2 database with the map number before passing to step 1155. Referring back to test step 1130, if the test step determines that the map level 2 database record has an updated map number, control is directly passed to step 1155 without invoking either step 1135 or step 1145. Referring back to step 1056, once the test of the map level 2 database determines that the record is past the end of the file, module 1050 is terminated and module 1160 is invoked.

Detailed Description Text (86):
Module 1160, detailed at FIG. 26, updates map level 2 database with the area map center point and creates the individual request map data file. Module 1160 is accessed from step 1056 of module 1050 at step 1161. Step 1 161 indexes the map level 1 database by map number, indexes the map level 2 database by map number, and links the map level 2 database to the map level 1 database on the map number before passing execution to step 1162. Step 1162 moves to the beginning of the map level 2 database. At step 1163, a test is made of the map level 2 database to determine that the current record is not past the end of file. If it is not past the end of the file, at step 1164 map level 2 database is updated to include: the item description, the map center coordinates from map level 1, description line 1, map scale, and box scale. After update step 1164 is complete, step 1200 accesses the next record in the map level 2 database. Step 1200 loops execution of the program back to step 1163. Once step 1163 determines that the current record is past the end of the file in map level 2 database, step 1201 is invoked. Step 1201 tests for a single record in the map level 1 database. If a single record exists in map level 1 database, step 1202 is invoked, which updates that database with: the map level; map center coordinates; description line 1, i.e., the phrase "Area Map"; description line 2, which is the area name; map scale; and box scale before closing all databases at step 1240. Step 1240 invokes step 1241 which: opens and initializes the map data database; merges records from map level 1, map level 2, and map level 3 databases into the map data database. Steps 1241 then updates the map data database with the map series; indexes the map data database on the map series, map level, and map number; copies data to the individual request map data file; and closes the map data database. After completing the foregoing actions, step 1241 invokes module 1250. Referring back to test step 1201, if multiple records are determined to exist in the map level 1 database, module 1250 is invoked directly.

Detailed Description Text (89):

Main processing loop 1290 is detailed at FIG. 28. Invoked by step 1285 of module 1260, step 1291 tests the individual request map data file for not past end of file. If the individual request map data file is not past the end of file, the individual map data file is read at step 1292 and the map series, map level, map number, map item, map center as latitude and longitude, the point as latitude and longitude, description lines 1 and 2, the map scale and box scale are all read. At step 1293, the map level is tested. If the map level is found to be 1, step 1294 is invoked, which copies description line 2 to the location description and invokes step 1295, which tests for a change in either the map level or the map number. Referring back to step 1293, if the map level is found not to be 1, step 1295 is again invoked. If the determination by step 1295 indicates that a change has been made in either the map level or the map number, pre-new map processing module 1300 is invoked, after which step 1400 is sequentially invoked. Step 1400 stores the current map level and map number to previous map level and map number, calculates and stores the map scale, both in terms of height and width, calculates and stores the map center coordinates as latitude and longitude, stores the level of map detail, calculates and stores the viewport window size, and calculates the physical location of the map on the page based on the map level and the map number, after which the draw map module 1500 is invoked. After execution of module 1500, the map level is tested at step 1600. Referring back to step 1295, if no change is determined to exist in either the map level or the map number, step 1600 is invoked directly. Step 1600 tests the map level as either a detail map or a general map or area map. If the map is a detail map, step 1650 is invoked, which calculates the physical location of the property dot and draws and labels the property dot. If step 1600 determines the map level to be either a general map level 1 or area map level 2, step 1601 calculates the physical location and scale of the outline box and draws and labels the outline box. After execution of either step 1601 or 1650, step 1690 is invoked, which moves to the next record in the individual request map data file. It returns control of the execution of this module to step 1291 as a loop. When step 1291 determines that the individual request map data file is past the end of the file, module 1700 is invoked and the main processing loop is exited.

Detailed Description Text (93):
After execution of step 1400, draw map module 1500 is invoked at step 1501. Draw map module 1500 is detailed at FIG. 32. Invoked by step 1400, this step sets the mapping parameters, including window, center, map range, level of detail, maximum number of line labels, label curvature, maximum area points, and maximum area bounds. Step 1501 then invokes step 1502, which sets mapping retrieval for area features at the most detailed level. Step 1503 then draws the area features before step 1520 sets the mapping retrieval for line features, i.e., roads. Step 1521 then sets the level of detail for drawing at the most detailed level. Step 1522 draws the line features before step 1530 sets the level of detail for drawing to the next level of detail, i.e., less detail. After setting the next level of detail, a test is made at step 1540 to determine if all levels of detail have been drawn. If more levels remain to be drawn, a loop is set up and step 1522 is again invoked. Once all levels have been drawn, step 1550 is invoked by step 1540, which labels the entire map and step 1580 turns off the PostScript.TM. clipping action before returning control to the main processing loop at step 1600.

Detailed Description Text (95):
Referring now to FIG. 33, an example of a listings sheet generated by the present invention is disclosed. Listings sheet 40 comprises header 41, area 42, target price 43 as input by user, price range factor 44, property location column 45, price column 46, bedroom column 47, date and time open column 48, and individual listings corresponding to map points 75.

Detailed Description Text (96):
Shown at FIG. 34 is a detail map 80, generated by the present invention, having on it one map point 75, centered on the detail map.

Detailed Description Text (97):
Shown at FIG. 35 is a second detail map 80 having on it several points 75 generated by the present invention.

Detailed Description Text (100):
One methodology for semi-automated geocoding of a real estate listings database, by a system operator, is shown at FIG. 38. As a preliminary to automated geocoding, the system operator, at step 1801, corrects the data provided by subscribers interested in listing their properties. After correcting any errors, the data is submitted to the automated portion of the geocoding process at step 1802 where the initial criteria defined by the geocoding strategy is set. This criteria defines the parameters that determine the degree of accuracy to be applied while geocoding addresses, e.g., search flags on each address component requiring either an exact match or allowing a wildcard match. At step 1803, the individual address components, i.e., address, street, city, postal ZIP code, and state, are validated against the Geolndex file. If each component is found to be valid or "good, the entire address combination is submitted for geocoding at step 1804. If the geocoding at step 1804 results in a single match between the address combination submitted and the Geolndex file, an assessment is made at step 1805 regarding the positional accuracy as determined by the geocoding strategy criteria. If the positional accuracy is acceptable, the coordinates are stored at step 1806. If the positional accuracy is not acceptable, a determination is made at step 1807 regarding whether or not the geocoding strategy has been exhausted. If it has not, the criteria are changed at step 1808 before looping back to test the components at step 1803. Referring back to step 1804, if the geocoding results in no match between the address combination submitted and the Geolndex file, step 1807 is directly invoked. Referring again back to step 1804, if the geocoding results in multiple matches between the address combination submitted and the Geolndex file, processing options are checked at step 1809. If automated processing options are available, step 1807 is again directly invoked. If only manual processing options are available, geocoding is performed manually at step 1810 and the resulting coordinates are stored. Referring back to the component test at step 1803, if any of the components is found to be invalid or in error, a determination is made at step 1811 as to the probable source of the problem. If the source of the problem is the GeoIndex file, adjustments for errors and/or omissions are made at step 1812 before looping back to test the components at step 1803. If the source of the problem at step 1811 is the submitted data, errors are corrected manually at step 1813 before looping back to test the components at step 1803. Referring back to step 1807, if the geocoding strategy has been exhausted, all defined approaches to geocoding have been unsuccessfully tried, then geocoding is performed manually at step 1810 on all remaining items that have not yet been geocoded.

Other Reference Publication (3):
A Status Report, "Front-Line Uses of GIS in Public and Private Sector Real Estate Today", Property Tax Journal, V12, N1, pp. 77-85, Mar. 1993.

CLAIMS:

3. The computer implemented process of claim 2, wherein said map page further includes a map point corresponding to a selected one of said listings, said map point correlating to a unique latitude and longitude, wherein said step of creating said map page further comprises the steps of:

defining said map point as said selected one of said listings; and

centering said map page around said map point.

4. The computer implemented process of claim 3, wherein said step of centering said map page further comprises the steps of:

generating an internal window corresponding to, and smaller than, said map page;

calculating the positional boundaries of said map page, using said internal window, available for adding a subsequent map point to said map page;

deriving the centerpoint of said map page from said map point; and

defining the centerpoint of said map page as the said latitude and said longitude of said map point.

5. The computer implemented process of claim 4, wherein said map page further includes a plurality of said map pages, said step of centering said map page further comprising the steps, for each of said subsequent map points, of:

determining if said subsequent map point can be added to said map page;

responsive to the determination, by said step of determining if said map point can be added to said map page, that a subsequent map point can be added to said map page, including said subsequent map point on said map page and recalculating said internal window centered on the midpoint between the latitudinal and longitudinal extremes of all the map points associated with said map page;

responsive to the determination, by said step of determining if said map point can be added to said map page, that said subsequent map point cannot be added to said map page, determining if said subsequent map point can be added to any previously defined map pages of Said plurality of map pages;

responsive to the determination, by said step of determining if said subsequent map point can be added to any previously defined map pages, that said subsequent map point can be added to at least one of said previously defined map pages, including said subsequent map point on said one of said previously defined map pages and recalculating said internal window of said one of said previously defined map pages centered on the midpoint between the latitudinal and longitudinal extremes of all the map points associated with said map page; and

responsive to the determination, by said step of determining if said subsequent map point can be added to said map page, that said subsequent map point cannot be added to at least one of said previously defined map pages, creating a subsequent map page, defining the centerpoint of said subsequent map page as said latitude and said longitude of said subsequent map point, and calculating the internal window of said subsequent map page centered on said subsequent map point.

6. The computer implemented process of claim 5, wherein said plurality of map pages further comprises a series of maps of decreasing scale, said series of maps including at least one each of a general map of small scale, an area map of intermediate scale, and a detail map of large scale, wherein said step of centering said map page around said map point further comprises the steps of:

defining a first alternate map point for display on said area map, said first alternate map point comprising the outline of said detail map; and

defining a second alternate map point for display on said general map, said second alternate map point comprising the outline of said area map.

8. The computer implemented process of claim 4, wherein said step of generating an internal window further comprises the steps of:

defining said map page size; and

defining a border buffer for said map page, said border buffer for defining a portion of said map page which, while displayed, contains none of said map points.

9. The computer implemented process of claim 3, wherein said step of creating said map page further comprises the steps of:

calculating the scale of said map page;

drawing said map page, displaying thereon said map point; and

labeling said map page.

10. For a programmed computer having memory, data storage and retrieval means, a plurality of listings stored in said data storage and retrieval means, each of said listings being defined in terms of geographic coordinates, a mapping database containing alphanumeric map data which define map features in terms of geographic coordinates stored in said data storage and retrieval means, display means, telephone receiving and transmitting means, and voice processing means, a computer implemented process for dynamically creating a map comprising a plurality of map pages including a series of maps of decreasing scale and a cover page summarizing information represented on said map, said series of maps including at least one each of a general map of small scale, an area map of intermediate scale, and a detail map of large scale, said plurality of maps created responsive to a remote user's request received on said telephone receiving means, and for transmitting said map to said user via said transmitting means, the process comprising the steps of:

responsive to prerecorded voice message prompts from said voice processing means, receiving said user's requests as a first DTMF telephonic signal via said telephone receiving means;

comparing said request with said listings;

selecting ones of said listings which most closely approximate said user's request;

dynamically creating said cover page as a graphical image using said selected ones of said listings;

storing said cover page as an image file in said data storage and retrieval means;

defining a map point for each selected one of said listings, said map point having a particular set of latitude and longitude coordinates, and correlating to one of said listings;

defining a first alternate map point for display on said area map, said first alternate map point comprising the outline of said detail map;

defining a second alternate map point for display on said general map, said second alternate map point comprising the outline of said area map;

for each said detail, area and general map pages;

a.) generating an internal window corresponding to, and smaller than, said map page, said internal window for calculating the positional boundaries of said map page available for adding a subsequent map point to said map page;

b.) generating a border buffer for said map page, said border buffer for defining a portion of said map page which, while displayed, contains none of said map points;

c.) generating the positional boundaries of said map page available for adding to said map page the appropriate one of said map point, said first alternate map point and said second alternate map point;

d.) deriving the centerpoint of each of said map pages from said map point, and;

e.) initially defining the centerpoint of each of said map pages as said latitude and said longitude of said map point;

f.) for each of said subsequent map points,

1.) determining if said subsequent map point can be added to said map page,

2.) responsive to the determination, by said step of determining if said map point can be added to said map page, that a subsequent map point can be added to said map page, including said subsequent map point on said map page and recalculating said internal window centered on the midpoint between the latitudinal and longitudinal extremes of all the map points associated with said map page,

3.) responsive to the determination, by said step of determining if said map point can be added to said map page, that said subsequent map point cannot be added to said map page, determining if said subsequent map point can be added to any previously defined map pages of said plurality of map pages,

4.) responsive to the determination, by said step of determining if said subsequent map point can be added to any previously defined map pages, that said subsequent map point can be added to at least one of said previously defined map pages, including said subsequent map point on said one of said previously defined map pages and recalculating said internal window of said one of said previously defined map pages centered on the midpoint between the latitudinal and longitudinal extremes of all the map points associated with said map page, and

5.) responsive to the determination, by said step of determining if said subsequent map point can be added to said map page, that said subsequent map point cannot be added to at least one of said previously defined map pages, creating a subsequent map page, defining the centerpoint of said subsequent map page as said latitude and said longitude of said subsequent map point, and calculating the internal window of said subsequent map page centered on said subsequent map point;

g.) calculating the scale of each said map pages,

h.) dynamically creating each of said map pages as graphical images using alphanumeric data from said mapping database, said map pages containing map information specific to said selected ones of said listings,

i.) labeling each of said map pages with said selected ones of said map points;

storing each of said map pages in said data storage and retrieval means as an image file; and

transmitting, as a second telephonic signal via said telephone transmitting means, said cover page and said map pages from said data storage and retrieval means to said user, in a format suitable for facsimile reception.